

# PATENT ABSTRACTS OF JAPAN

(11)Publication number : 07-084943

(43)Date of publication of application : 31.03.1995

(51)Int.Cl.

G06F 13/38

H04L 7/04

H04L 29/08

(21)Application number : 05-232090

(71)Applicant : MITSUBISHI ELECTRIC CORP

(22)Date of filing : 17.09.1993

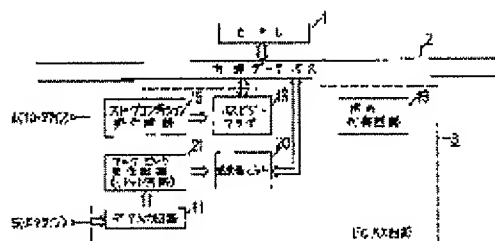
(72)Inventor : ISHII ETSUKO  
HOSOYA OSAMU

## (54) SERIAL COMMUNICATION DEVICE

### (57)Abstract:

**PURPOSE:** To obtain a device which prevents plural devices from generating communication start signals repeatedly and enables a CPU to easily recognize that the communication start signals are prevented from being generated by equipping the device with a start condition inhibiting circuit and a transmission/ reception bit and a master/slave bit which are reset by the start condition inhabiting circuit.

**CONSTITUTION:** When a communication is already started, the start condition inhabiting circuit 28 inhibits a start condition generating circuit 14 from generating a communication start signal. When the start condition inhabiting circuit 28 inhibits the communication start signal from being generated, the transmission/ reception bit 20 and master/slave bit 27 are reset to a low level at the same time. The CPU 1 reads in the transmission/reception bit 20 and master/ -slave bit 27 and detects the resetting of them to the low level to recognize that the start condition inhibiting circuit 28 inhibits the communication start signal from being generated.



(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平7-84943

(43) 公開日 平成7年(1995)3月31日

(51) Int.Cl. <sup>6</sup>	識別記号	序内整理番号	F I	技術表示箇所
G 0 6 F 13/38	3 5 0	8944-5B		
H 0 4 L 7/04	B	7741-5K		
29/08		9371-5K	H 0 4 L 13/ 00	3 0 7 Z

審査請求 未請求 請求項の数4 O L (全 9 頁)

(21) 出願番号 特願平5-232090

(22) 出願日 平成5年(1993)9月17日

(71) 出願人 000006013

三菱電機株式会社

東京都千代田区丸の内二丁目2番3号

(72) 発明者 石井 悦子

伊丹市瑞原4丁目1番地 三菱電機株式会  
社北伊丹製作所内

(72) 発明者 細谷 理

伊丹市瑞原4丁目1番地 三菱電機株式会  
社北伊丹製作所内

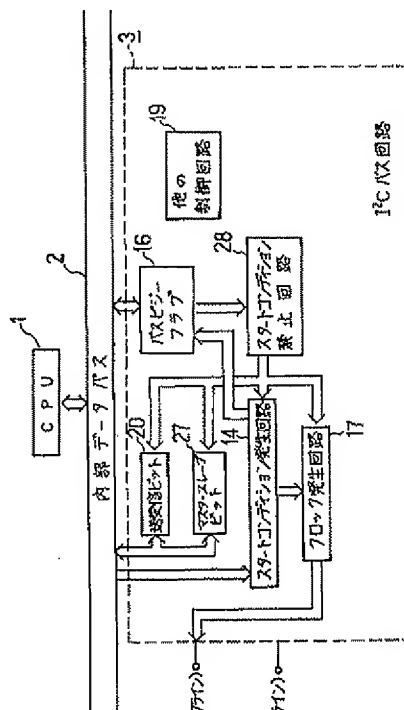
(74) 代理人 弁理士 田澤 博昭 (外1名)

(54) 【発明の名称】 シリアル通信装置

(57) 【要約】

【目的】 複数デバイスによる通信開始信号の重複発生を防止し、通信開始信号の発生を禁止されたことをCPUが容易に認識することのできるシリアル通信装置を得る。

【構成】 他のマスタデバイスより通信開始信号が発生されていることを認識して、自分自身による通信開始信号の発生を禁止するスタートコンディション禁止回路と、このスタートコンディション禁止回路によってリセットされる送受信ビット、およびマスタ・スレーブビットを備える。



## 【特許請求の範囲】

【請求項1】 複数のデバイスをクロックラインとデータラインによって接続し、前記デバイス中の1つが通信に際してクロックを発生させるマスタデバイスとなり、他が前記マスタデバイスの発生するクロックに同期してデータ通信を行うスレーブデバイスとなって、アドレスデータおよび制御データを伝送するシリアル通信装置において、前記マスタデバイスが前記スレーブデバイスに対して確認応答信号を生成しない場合に、前記確認応答信号を受信しないことを検出すると、データ通信の方向を決定する送受信ビットをリセットするリセット回路を設けたことを特徴とするシリアル通信装置。

【請求項2】 複数のデバイスをクロックラインとデータラインによって接続し、前記デバイス中の1つが通信に際してクロックを発生させるマスタデバイスとなり、他が前記マスタデバイスの発生するクロックに同期してデータ通信を行うスレーブデバイスとなって、アドレスデータおよび制御データを伝送するシリアル通信装置において、データの受信を行っている前記デバイスからデータの送信を行っている前記デバイスにデータ受信完了を通知するための確認応答信号の生成を、アックビット切り換えフラグに従って行うアックビット発生回路と、前記マスタデバイスとなった場合に確認応答信号を生成するためのアッククロックの前記クロックラインへの発生を、アッククロック発生フラグに従って行うアッククロック発生回路とを設けたことを特徴とするシリアル通信装置。

【請求項3】 複数のデバイスをクロックラインとデータラインによって接続し、前記デバイス中の1つが通信に際してクロックを発生させるマスタデバイスとなり、他が前記マスタデバイスの発生するクロックに同期してデータ通信を行うスレーブデバイスとなって、アドレスデータおよび制御データを伝送するシリアル通信装置において、前記マスタデバイスとなっている他のデバイスによって既に通信が開始されている時に通信開始信号を発生させようとした場合、前記通信開始信号の発生を禁止すると同時に、前記クロックの発生も禁止するスタートコンディション禁止回路を設けたことを特徴とするシリアル通信装置。

【請求項4】 前記スタートコンディション禁止回路が、前記通信開始信号および前記クロックの発生を禁止した時に、データ通信の方向を決定する送受信ビット、およびマスタデバイスとなってデータ通信に際して前記クロックを発生させることが可能か否かを決定するマスタ・スレーブビットも同時にリセットする機能を備えていることを特徴とする請求項3に記載のシリアル通信装置。

## 【発明の詳細な説明】

【0001】

データ通信をバスラインを介して行い、そのデバイス中の1つがマスタデバイス、他がスレーブデバイスとなって、アドレスデータおよび制御データの伝送を行うシリアル通信装置に関するものである。

## 【0002】

【従来の技術】このようなマルチマスタのバス方式を備えたシリアル通信装置としては、従来よりインター・アイシー・バス（Inter I C Bus；以下 $I^2C$ バスという）が知られている。図5はこの $I^2C$ バスによる従来のシリアル通信装置について、それを構成する1つのデバイスを示したブロック図である。図において、1は当該デバイスの全体制御を行う中央演算処理装置（以下CPUという）であり、2は当該デバイス内の内部データバス、3はこの内部データバス2を介してCPUとデータの授受を行っている $I^2C$ バス回路である。4はこのような各デバイス間のデータ通信に際してクロックが伝送されるクロックラインであり、5は同じく、アドレスデータおよび制御データが伝送されるデータラインである。

【0003】また、 $I^2C$ バス回路3内において、10はデータライン5にアドレスデータや制御データを送信するデータ出力回路であり、11はデータライン5よりアドレスデータや制御データを受信するデータ入力回路である。12はデータを受信しているデバイス（以下受信デバイスという）にて、データを送信しているデバイス（以下送信デバイスという）に対して1バイトのデータの受信終了直後にデータライン5をローレベルにして、確認応答信号であるアック（ACK；Acknowledge）ビットを発生するアックビット発生回路であり、13は送信デバイスにて、受信デバイスからの前記アックビットを受信するアックビット受信回路である。14は通信開始信号を発生するスタートコンディション発生回路、15は通信終了信号を発生するストップコンディション発生回路であり、16は通信開始信号の発生でハイレベル、通信終了信号の発生でローレベルとなり、クロックライン4およびデータライン5が他のデバイスによって使用されていて使用不可能な状態、即ちバスビジー（Bus-busy）状態にあるか否かを示すバスビジーフラグである。17は当該デバイスがマスタデバイスとして動作している時にクロックライン4に送出するクロックを発生するクロック発生回路であり、18はマスタデバイスが1バイトのデータ通信直後にクロックラインに送出するアッククロックを発生するアッククロック発生回路である。19はこれら以外にこのデバイス内に備えられた制御回路を総括的に示した他の制御回路である。

【0004】次に動作について説明する。今、あるデバイスがマスタデバイスとなり、他のスレーブデバイスと通信を行おうとした時、まずスタートコンディション発生回路14で通信開始信号を発生する時に、バスビジー

判断する。その結果、バスフリー状態であることが確認されると通信開始信号を発生させ、この通信開始信号が発生したことを受けてバスビジーフラグ16はハイレベルにセットされる。クロック発生回路17により1<sup>1</sup>Cバス回路3内でクロックが発生すると、そのクロックはクロックライン4を介してスレーブデバイスに伝えられ、データ通信が開始される。また、データ通信を終了する場合は、マスタデバイスがストップコンディション発生回路15により通信終了信号を発生させ、クロックライン4およびデータライン5をハイレベルにしてバスフリー状態とする。この時バスビジーフラグ16はリセットされ、他のデバイスがマスタデバイスとなって通信を行うことが可能となる。

【0005】マスタデバイスは1バイトのデータ通信に必要なクロックをクロック発生回路17より出力した直後に必ず、アッククロック発生回路18よりアッククロックを発生させてそれをクロックライン4に送出する。このアッククロックに同期してアドレスデータおよび制御データを受信した受信デバイスは、データライン5をローレベルにしてそのデータの受信確認応答信号であるアックビットを返送する。このアックビットは、受信デバイスが1バイトのデータを受信することができた場合にのみ、そのアックビット発生回路12より発生されるものである。送信デバイスではこのアックビットをアックビット受信回路13で受信して、データがきちんと通信できていることを確認した後、次の処理を開始する。

【0006】アッククロック発生時にデータライン5がハイレベルに保たれていた場合、送信デバイスはそれによってデータ通信がきちんと行われなかったことを認識し、マスタデバイスはストップコンディション発生回路15より通信終了信号を発生させようとする。このとき、マスタデバイスが送信デバイスであれば、そのまま通信終了信号を発生することができるが、スレーブデバイスが送信デバイスであった場合には、アックビットを受け取らなかったときにソフトウェアによってスレーブデバイスのデータ通信方向を変更し、それを受信デバイスに変えてからマスタデバイスが通信終了信号を発生する。

【0007】また、マスタデバイスが受信デバイスであるときに通信終了信号を出力したい場合には、マスタデバイスはアッククロック発生時にデータ通信の方向をソフトウェアで変更して送信デバイスとなり、1ビット分のデータとしてデータライン5をハイレベルに保つ。アッククロック発生時にデータライン5がハイレベルに保たれていると、スレーブデバイスはCPU1においてアックビットが返らなかったことを認識する。その後、送信デバイスであったスレーブデバイスは、ソフトウェアによってデータ通信方向をデータ受信に変更するなど、通信終了信号を発生して次の処理を行う。

【発明が解決しようとする課題】従来のシリアル通信装置は以上のように構成されているので、通信を開始する場合に、バスシステムが他のマスタデバイスによって使用されているか否かを確認しなければならず、バスビジーフラグ16を常に読み込むことが必要であり、また、送信デバイスは受信デバイスからのアックビットの返送を常にCPU1で読み込んで次の処理を行う必要があるなどのため、ソフトウェア効率が悪くなるばかりか、通信開始信号を出力したにも関わらず、他のマスタデバイスによる通信が既に開始されていて当該通信開始信号が不成立となった場合に、それを認識する手段がないためそのまま通信を続けることとなり、さらに、意図的にアックビットを返送したくない場合には、ソフトウェアによってデータ通信の方向を変更してアックビット発生回路12をオフとした後、アッククロックと同期させてハイレベルのデータを送信させ、データライン5をハイレベルに保っておく必要があるため、ソフトウェアの負荷が増大するなどの問題点があった。

【0009】請求項1に記載の発明は上記のような問題点を解消するためになされたもので、アックビット受信回路によってアックビットが返ってきたか否かの判断をし、その後通信終了信号、通信再開信号の発生等、マスタデバイスからの処理がスムーズに行えるシリアル通信装置を得ることを目的とする。

【0010】また、請求項2に記載の発明は、マスタデバイスが1バイトのデータ通信直後にアッククロックを発生させるか否かをフラグの切換で選択することができ、さらに、アッククロック発生時に受信デバイスがデータラインをローレベルにしてアックビットを送信デバイスに返すか、データラインをハイレベルに保ってアックビットを送信デバイスに返さないかを、フラグの切り換えによって容易に選択することができるシリアル通信装置を得ることを目的とする。

【0011】また、請求項3に記載の発明は、CPUによってバスビジーフラグを常に読み込む必要がなく、任意に通信開始信号を発生することができ、既に他のマスタデバイスにてバスシステムが利用されている場合にはその通信開始信号の発生を自動的に禁止して、複数のデバイスが同時にマスタデバイスとなって通信を行うことを防止できるシリアル通信装置を得ることを目的とする。

【0012】また、請求項4に記載の発明は、上記請求項3に記載の発明によって送信開始信号の発生が禁止された場合に、そのことをCPUが容易に認識できるシリアル通信装置を得ることを目的とする。

【0013】

【課題を解決するための手段】請求項1に記載の発明に係るシリアル通信装置は、マスタデバイスがスレーブデバイスに対してデータ通信を行う際に、データライン5をハイレベルに保つ。アッククロック発生時にデータライン5がハイレベルに保たれていると、スレーブデバイスはCPU1においてアックビットが返らなかったことを認識する。その後、送信デバイスであったスレーブデバイスは、ソフトウェアによってデータ通信方向をデータ受信に変更するなど、通信終了信号を発生して次の処理を行う。

と、データ通信の方向を決定する送受信ビットのリセットを行うリセット回路を設けたものである。

【0014】また、請求項2に記載の発明に係るシリアル通信装置は、アックビット切り換えフラグに従って受信デバイスから送信デバイスへの確認応答信号の生成を行うアックビット発生回路、およびアッククロック発生フラグに従ってマスタデバイスとなった場合のアッククロックの発生を行うアッククロック発生回路を設けたものである。

【0015】また、請求項3に記載の発明に係るシリアル通信装置は、他のデバイスがマスタデバイスとなって既に通信が行われている時に通信開始信号を発生させようとした場合、その通信開始信号の発生を禁止すると同時に、クロックの発生も禁止するスタートコンディション禁止回路を設けたものである。

【0016】また、請求項4に記載の発明に係るシリアル通信装置は、前記スタートコンディション禁止回路にさらに、通信開始信号およびクロックの発生禁止時に、送受信ビットのリセットと、マスタデバイスとなってデータ通信に際してクロックを発生させることが可能か否かを決定するマスタ・スレーブビットのリセットも同時に行う機能を持たせたものである。

#### 【0017】

【作用】請求項1に記載の発明におけるリセット回路は、スレーブデバイスが送信デバイスとなってデータを送信した際、アッククロックの発生時にマスタデバイスより確認応答信号が返送されてこなかった場合に、データ通信の方向を決定する送受信ビットをリセットすることにより、ソフトウェアによって確認応答信号が返ってきたか否かを判断する必要がなくなり、また、ソフトウェアで送受信ビットを変える必要もなくなって、ソフトウェア効率の向上がはかれるシリアル通信装置を実現する。

【0018】また、請求項2に記載の発明におけるアックビット発生回路は、受信デバイスから送信デバイスへの確認応答信号をアックビット切り換えフラグに従って生成し、アッククロック発生回路はマスタデバイスとなった場合に、アッククロックをアッククロック発生フラグに従って発生することにより、意図的に確認応答信号を返したくない場合には、フラグ切り換えを行うだけでそれを可能とし、確認応答信号が不要な場合も、マスタデバイスがアッククロックを発生させるか否かをフラグ切り換えだけで容易に選択することが可能なシリアル通信装置を実現する。

【0019】また、請求項3に記載の発明におけるスタートコンディション禁止回路は、他のマスタデバイスにて既に通信が行われている時には、通信開始信号の発生を禁止すると同時に、クロックの発生も禁止することにより、バフビジーフラグをクロックで常にエスケープして

に、1つのマスタデバイスのみがクロックラインおよびデータラインを使用したデータ通信を行うようにして、通信中のマスタデバイスが他のマスタデバイスによって影響を受けることのないシリアル通信装置を実現する。

【0020】また、請求項4に記載の発明におけるスタートコンディション禁止回路は、さらに、通信開始信号およびクロックの発生禁止時に、送受信ビットのリセットとマスタ・スレーブビットのリセットも同時に行うことにより、この送受信ビットとマスタ・スレーブビットの読み込みによって通信開始信号の発生が禁止されたことが容易に認識でき、以降の処理がスムーズに行えるシリアル通信装置を実現する。

#### 【0021】

##### 【実施例】

実施例1. 以下、この発明の実施例1を図について説明する。図1は請求項1に記載した発明の一実施例を示すブロック図である。図において、1はCPU、2は内部データバス、3はI<sup>2</sup>Cバス回路、4はクロックライン、5はデータライン、11はデータ入力回路、15はストップコンディション発生回路、16はバスビジーフラグ、19は他の制御回路であり、図5に同一符号を付した従来のそれらと同一、もしくは相当部分であるため詳細な説明は省略する。また、20は当該デバイスが送信デバイスである場合にはハイレベルとなり、受信デバイスである場合にはローレベルとなってデータ通信の方向を決定する送受信ビットである。21はマスタとなっているデバイスがスレーブとなっているデバイスに対してアックビットを生成しない場合に、当該アックビットが受信されないことを検出して送受信ビット20をリセットするリセット回路としても機能する点で、図5に符号13を付した従来のものとは異なったアックビット受信回路（リセット回路）である。

【0022】次に動作について説明する。今、あるデバイスがマスタデバイスとなって、スレーブデバイスにアドレスデータと制御データを送信している場合、1バイトのデータ通信直後にマスタデバイスはクロックライン4上にアッククロックを発生させる。マスタデバイスから送信された1バイトのデータがちゃんと受信できたとき、スレーブデバイスはデータ受信確認応答としてアッククロック発生時にデータライン5をローレベルにする。このアックビットが返ってきたことがアックビット受信回路21で検出されると、マスタデバイスはスレーブデバイスで1バイトのデータがきちんと受信できたものと認識して、次のデータ、または通信終了信号、通信再開信号を送信する。一方、アッククロック発生時にデータライン5がハイレベルのままでは保たれていて、アックビットが返送されてこなかったことがアックビット受信回路21で検出されると、マスタデバイスはスレーブデバイスで1バイトのデータがちゃんと受信できたか

5によって通信終了信号を発生させる等の処理を実行する。なお、この通信終了信号の発生に伴って、バスビジーフラグ16はリセットされる。

【0023】また、あるスレーブデバイスが送信デバイスとなり、マスタデバイスに制御データを送信している場合には、スレーブデバイスの送受信ビット20がハイレベルとなり、マスタデバイスの送受信ビット20はローレベルとなる。1バイトのデータ通信が終了した直後に、マスタデバイスがクロックライン4上にアッククロックを発生させ、データ通信がきちんと行われた場合には、マスタデバイスがデータライン5をローレベルにしてスレーブデバイスに対して確認応答信号であるアックビットを返送する。スレーブデバイスはこのマスタデバイスより送られてきたアックビットを受けると、次の制御データを送信する処理を実行する。

【0024】一方、アッククロック発生時にアックビット受信回路21によってアックビットが受信できなかったスレーブデバイスは、データ通信がきちんと行われていなかったと認識して、マスタデバイスからの通信終了信号等を受信できるように送受信ビット20をローレベルにリセットする。なお、この送受信ビット20のリセットは、この実施例1のようにアックビット受信回路21で行うものの外、独立に設けたリセット回路によって行うようにしてもよい。この送受信ビット20のリセットが行われたスレーブデバイスでは、他のスレーブデバイスと同様に通信終了信号や通信再開信号を受け取ることができる状態となり、その後、マスタデバイスからの通信データを受信する。また、この送受信ビットがリセットされたことにより、CPU1もデータ通信がきちんと行われず、アックビットが返ってこなかったことを認識することができる。

【0025】実施例2. 次に、この発明の実施例2を図について説明する。図2は請求項2に記載した発明の一実施例を示すブロック図であり、相当する部分には図5に示した従来のものと同一の符号を付してその説明を省略する。図において、22は受信デバイスから送信デバイスにデータ受信完了を伝えるために生成される、確認応答信号としてのアックビットを発生させるか否かを切り換えによって選択するアックビット切り換えフラグであり、23はこのアックビット切り換えフラグ22に従って、受信デバイスから送信デバイスへのアックビットの生成を行う点で、図5に符号12付した従来のものとは異なるアックビット発生回路である。24は1バイトのデータ通信の直後に発生するアックビットを生成するためのアッククロックを、マスタデバイスがクロックライン4上に発生させるか否かを選択するアッククロック発生フラグであり、25はこのアッククロック発生フラグ24に従ってアッククロックの発生を行う点で、図5に符号10付した従来のものとは異なるアッククロ...

【0026】次に動作について説明する。今、アッククロック発生フラグ24がハイレベルとなっているものとする。この場合、1バイトのデータ通信直後にアックビット用のアッククロックをアッククロック発生回路25より発生させ、クロック発生回路17を通じてクロックライン4に送出する。逆に、アッククロック発生フラグ24がローレベルとなっている場合には、アッククロック発生回路25がリセットされてクロックライン4上にはアッククロックは送出されない。以上のように、アッククロック発生フラグ24の切り換えによってアッククロックを発生させるか否かを選択することができる。

【0027】また、アックビット切り換えフラグ22が、例えばハイレベルとなっているときにクロックライン4上にアッククロックが発生した場合、アックビット発生回路23が作動して、データ出力回路10を通してデータライン5をローレベルにする。逆に、アックビット切り換えフラグ22がローレベルとなっているときにクロックライン4上にアッククロックが発生した場合には、アックビット発生回路23がリセットされて、データ出力回路10からはデータは出力されず、データライン5はハイレベルに保たれる。

【0028】実施例1で説明した、スレーブデバイスがデータを送信している場合にマスタデバイスが通信を終了したいときには、アックビット切り換えフラグ22をローレベルにして、アッククロック発生時にデータライン5をハイレベルのままに保つ。データを送信しているスレーブデバイスはアックビット受信回路13でそれを検出して、アックビットが返送されてこなかったと認識する。よって、データを受信しているマスタデバイスがストップコンディションを発生するなど、データ通信方向を変更する必要がある場合には、マスタデバイスはアックビット切り換えフラグ22をローレベルにしてアックビットを返送しないようにする。

【0029】実施例3. 次に、この発明の実施例3を図について説明する。図3は請求項3に記載した発明の一実施例を示すブロック図であり、相当する部分には図5に示した従来のものと同一の符号を付してその説明を省略する。図において、26はマスタデバイスとなっている他のデバイスによって既に通信が開始されている時に、スタートコンディション発生回路14より通信開始信号を発生させようとした場合、当該通信開始信号の発生を禁止すると同時に、クロック発生回路17によるクロックの発生も禁止するスタートコンディション禁止回路である。

【0030】次に動作について説明する。ここで、他のマスタデバイスによってこのシステムバスが使用されている場合にはバスビジーフラグ16はハイレベルになっている。スタートコンディション禁止回路26はこのバスビジーフラグ16がハイレベルになっている時、フラ...



生が試みられた場合に、当該通信開始信号の発生を禁止する。これにより、当該バスシステムが他のマスタデバイスで既に使用されているにも関わらず、スタートコンディションが重複して発生されることがなくなり、複数のマスタデバイスによってバスシステムが同時に使用されることを防止する。

【0031】このスタートコンディション発生回路14による送信開始信号の発生が禁止されると同時に、クロック発生回路17もスタートコンディション禁止回路26によってリセットされ、クロックライン4へのクロックの発生が禁止される。従って、この通信開始信号の発生に失敗したデバイスはスレーブデバイスとなり、クロックライン4を介して、既に通信を開始しているマスタデバイスからのクロックを受信し、スレーブデバイスとしてデータ通信を行う。

【0032】実施例4、次に、この発明の実施例4を図について説明する。図4は請求項4に記載した発明の一実施例を示すブロック図であり、相当する部分には図5に示した従来のものと同一の符号を付してその説明を省略する。図において、20は実施例1において説明した送受信ビットであり、27は当該デバイスがマスタデバイスとなってデータ通信を行うに際して、クロックを発生させることが可能か否かを決定するマスタ・スレーブビットである。28は通信開始信号とクロックの発生を禁止した時に、同時に送受信ビット20およびマスタ・スレーブビット27のリセットも行う機能を備えている点で、図2に符号26を付した実施例3のものとは異なったスタートコンディション禁止回路である。

【0033】次に動作について説明する。ここで、上記実施例3においては、スタートコンディション禁止回路26によって通信開始信号の発生が禁止された場合、そのことをCPU1が認識できず、そのままデータ通信処理を実行してしまうことになる。この実施例4は、スタートコンディション禁止回路28の作動によって通信開始信号を送信できなかったことを、CPU1が容易に認識できるようにしたものである。

【0034】今、送受信ビット20は例えば、アドレスデータおよび制御データを送信する送信デバイスではハイレベルとなり、それらを受信する受信デバイスではローレベルとなるというように定められている。また、マスタ・スレーブビット27は例えば、マスタデバイスとなってデータ通信に必要なクロックをクロックライン4上に送出しているデバイスではハイレベルとなり、スレーブデバイスとなってマスタデバイスが送出したクロックに同期してデータライン5上に発生するデータを送受信するデバイスではローレベルとなるというように定められている。

【0035】スタートコンディション発生回路14によ

のスレーブデバイスに送信するので、その送受信ビット20およびマスタ・スレーブビット27はともにハイレベルとなる。上記実施例3で説明したようなスタートコンディション重複禁止作用により、スタートコンディション禁止回路28にて通信開始信号の発生が禁止された場合、同時に送受信ビット20とマスタ・スレーブビット27がローレベルにリセットされる。この送受信ビット20およびマスタ・スレーブビット27がリセットされることにより、当該デバイスはスレーブデバイスとなってマスタデバイスからのデータの受信が可能となる。

【0036】CPU1はこれら送受信ビット20とマスタ・スレーブビット27を読み込んで、それらがローレベルにリセットされたことを検出することにより、スタートコンディション禁止回路28によって通信開始信号の発生が禁止されたことを認識することができる。

【0037】

【発明の効果】以上のように、請求項1に記載の発明によれば、スレーブデバイスが制御データを送信している場合にマスタデバイスより確認応答信号が返ってこない時、送受信ビットのリセットを行うように構成したので、当該スレーブデバイスはマスタデバイスからのデータ、および通信終了信号、通信再開信号の受信が可能となり、確認応答信号が返ってきたか否かをソフトウェアによって常にモニタしている必要性がなくなり、また、確認応答信号が返ってこなかったことを認識した後、ソフトウェアで送受信ビットを変える必要もなくなって、ソフトウェアの負荷が軽減され、ソフトウェア効率の向上がはかれる効果がある。

【0038】また、請求項2に記載の発明によれば、アックビット切り換えフラグに従って受信デバイスから送信デバイスへの確認応答信号を生成し、アッククロック発生フラグに従ってマスタデバイスとなった場合のアッククロックを発生するように構成したので、通信終了直後に確認応答信号を発生させるか否かの選択を、アッククロック発生フラグの切り換えを行うだけで簡単に行うことができ、また、アッククロック発生時に確認応答信号を返すか否かの選択も、アックビット切り換えフラグの切り換えだけで簡単に行えるため、意図的に確認応答信号を返さない場合に、1ビット分のアッククロック発生時にデータ通信の方向を変更してハイレベルのデータを確認応答信号として返送するという処理が不要となって、ソフトウェアの効率が向上するという効果がある。

【0039】また、請求項3に記載の発明によれば、他のマスタデバイスにて既に通信が行われている時には、通信開始信号の発生とクロックの発生を同時に禁止するように構成したので、バスビジーフラグを読み込まずに通信開始信号を発生させることが可能となり、複数のデバイスからの通信開始信号が重複した場合には、最初に通信開始信号の発生を試みたデバイスのみがマスタ

バイスとなるため、選択されたマスタデバイスの通信を妨げることはなく、従って、CPUがバスビジーフラグを常にモニタしている必要性がなくなり、ソフトウェア効率が向上する効果がある。

【0040】また、請求項4に記載の発明によれば、通信開始信号の発生を禁止されてマスタデバイスとして通信を行うことができなかったデバイスでは、送受信ビットとマスタ・スレーブビットも同時にローレベルにリセットされるように構成したので、それらを読み込んだCPUは通信開始信号の発生が禁止されたことを容易に認識することが可能となり、この受信ビットとマスタ・スレーブビットがハイレベルに維持できたと認識したCPUはその後の通信に必要な処理を行うが、それらがローレベルであればスレーブデバイスとしてマスタデバイスからのデータを受信し、マスタデバイスのデータ通信の処理を妨げないなどの処理を、この送受信ビットとマスタ・スレーブビットを読み込むことによって、確実に選択できる効果がある。

#### 【図面の簡単な説明】

【図1】この発明の実施例1によるシリアル通信装置を示すブロック図である。

\*

\*【図2】この発明の実施例2によるシリアル通信装置を示すブロック図である。

【図3】この発明の実施例3によるシリアル通信装置を示すブロック図である。

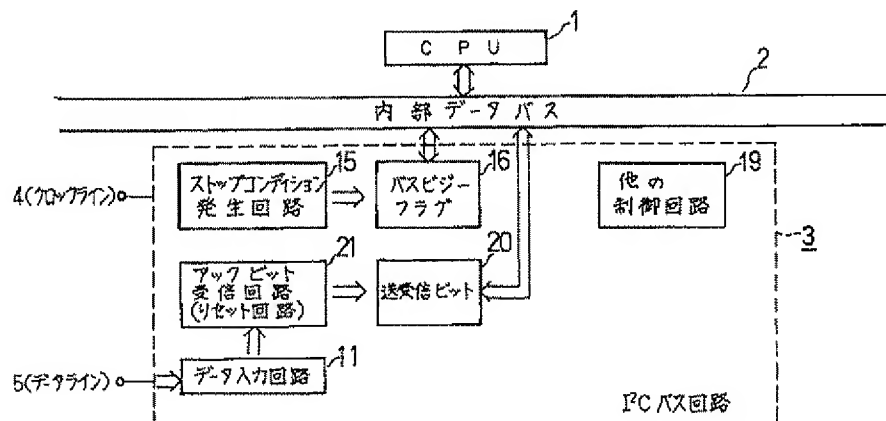
【図4】この発明の実施例4によるシリアル通信装置を示すブロック図である。

【図5】従来のシリアル通信装置を示すブロック図である。

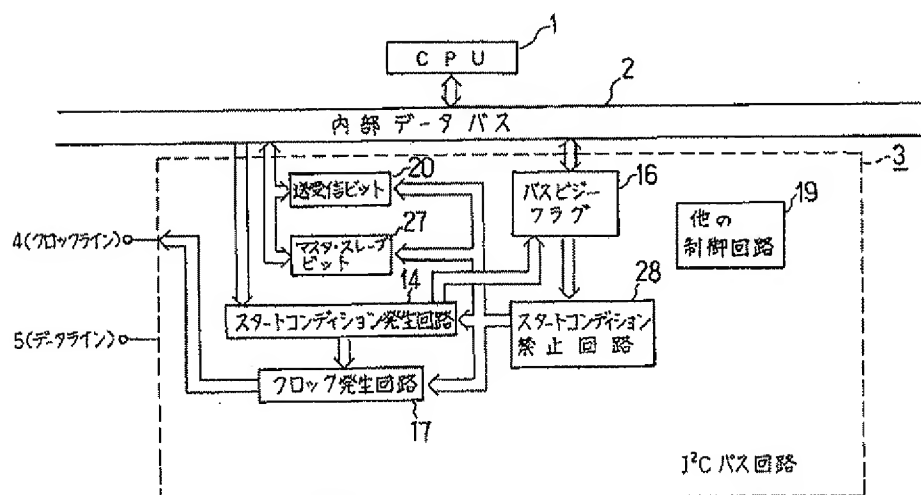
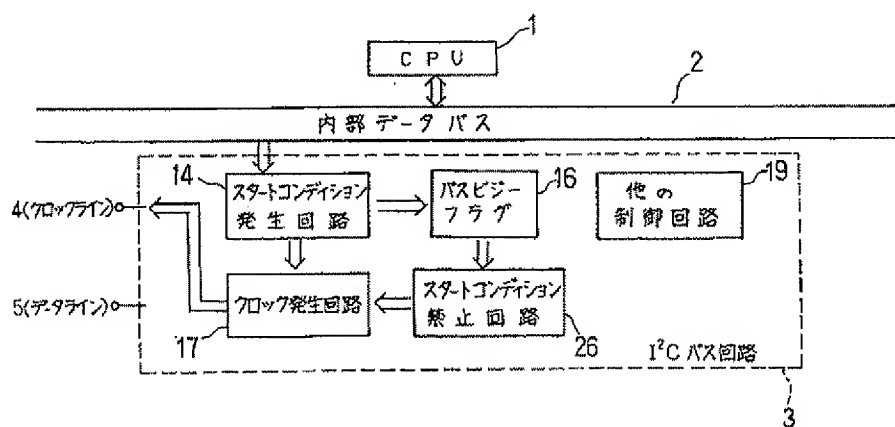
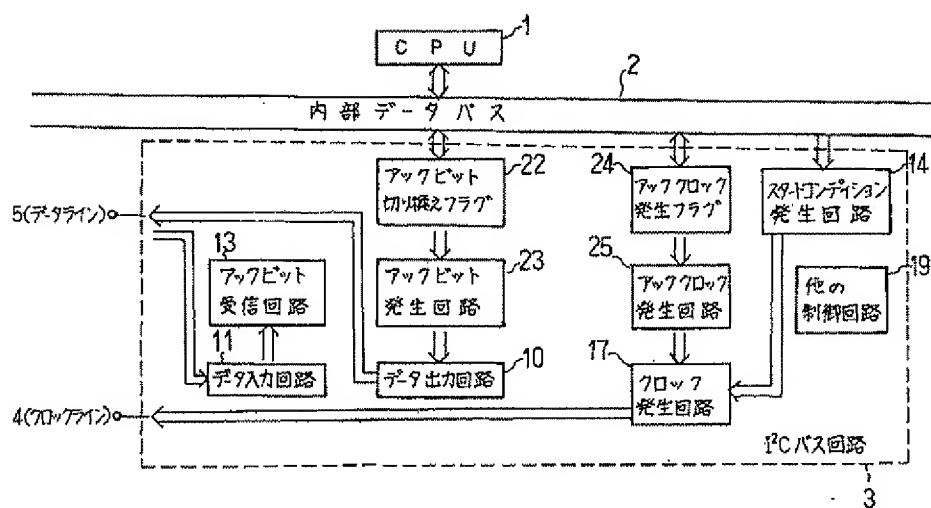
#### 【符号の説明】

- 4 クロックライン
- 5 データライン
- 20 送受信ビット
- 21 アックビット受信回路（リセット回路）
- 22 アックビット切り換えフラグ
- 23 アックビット発生回路
- 24 アッククロック発生フラグ
- 25 アッククロック発生回路
- 26 スタートコンディション禁止回路
- 27 マスタ・スレーブビット
- 28 スタートコンディション禁止回路

【図1】







【図5】

